User Interface Descriptions for SYM_Dsd

Interface Details	*Description*
Interface Description	Request by other SYM function or CCS subsystem for state data
Level 2 Interface Name	G_GUI_StateDataRequest
Level 3 Interface Name	G_GUI_StateDataRequest
Level 2 Source Process	GUI
Level 2 Destination Process	SYM_DistributeStateData
Level 3 Source/Destination Process	SYM_ManageStateDataRequest
Interface Elements	Refer to DSD Interface Descriptions, Dsd_flow.doc, for details
Operational Modes	All
Implications to GUI/CCSU	The CCS User will need to select a set of states to display; therefore, the GUI will need to provide the ability to create/modify/delete such sets and to convert such sets to the proper packet format.
Implications to SYM_xxx	
Interface Description	Return of data to requesting SYM function or CCS subsystem
Level 2 Interface Name	G_GUI_StateDataResponse
Level 3 Interface Name	G_GUI_StateDataResponse
Level 2 Source Process	SYM_DistributeStateData
Level 2 Destination Process	GUI
Level 3 Source/Destination Process	SYM_ManageStateDataRequest
Interface Elements	Refer to DSD Interface Descriptions, Dsd_flow.doc, for details
Operational Modes	All
Implications to GUI/CCSU	
Implications to SYM_xxx	The current plan is to send all the data available for each requested state. The API provided to the requesting clients must contain a set of methods allowing the client to extract only that information which is desired.

Interface Description	Start/Stop DSD Request - GUI will use an API method which builds a DSD on the Source Node
Level 2 Interface Name	N/A
Level 3 Interface Name	N/A
Level 2 Source Process	GUI
Level 2 Destination Process	N/A
Level 3 Source/Destination Process	N/A
Interface Elements	Source Data - Node (int) Subsystem (int) Process (int)
Operational Modes	All
Implications to GUI/CCSU	
Implications to SYM_xxx	Must provide the API described above
Interface Description	Start/Stop DSDResponse
Level 2 Interface Name	N/A
Level 3 Interface Name	N/A
Level 2 Source Process	SYM_DistributeStateData
Level 2 Destination Process	GUI
Level 3 Source/Destination Process	
Interface Elements	DSD Identifier - passed to processes who need to communicate with this DSD so they can use it in all future transmissions
Operational Modes	All
Implications to GUI/CCSU	GUI will need to provide the DSD ID to other processes as required - this implies that DSD must be started first ***OR*** the other processes must have an API with GUI to accept the DSD ID.
Implications to SYM_xxx	

Interface Description	Modify States - for testing purposes only!!
Level 2 Interface Name	
Level 3 Interface Name	
Level 2 Source Process	GUI
Level 2 Destination Process	SYM_DistributeStateData
Level 3 Source/Destination Process	SYM_InputStateData
Interface Elements	Header Data • Update Type (2 bits) 1) Set for one time only 2) Set and retain until further notification 3) Release from test and resume normal processing 4) Clear all test values • Number of test update records (int) For update types 1-3, the state list must be provided as a set of test update records. Types 1 and 2 will have all the information described below; type 3 will have only the State IDs State ID (int) Data to update (16 bits) - one bit per each item in True State, Expected State, and Compare Status, to indicate which specific fields are to be updated. True State (structure) • Spacecraft Time • True State Value - Raw • True State Value - EU • Flags (see FOF for bit descriptions) • Data Source (see FOF for bit descriptions) • Telemetry Format (see FOF for format codes) • Delta Value Expected State (structure) • Time Computed • Expected State Value • Expected State Format - Raw or EU • Expected State Tolerance (will be 0 for discretes) Compare Status (structure) • Time of last compare • Result of last compare - TS = ES± Tolerance? Yes/No
Operational Modes	All
Implications to GUI/CCSU	The CCS User will need to select a set of states to update; therefore, the User Interface may need to provide the ability to create/modify/delete such sets and to convert such sets to the proper packet format.
Implications to SYM_xxx	Need to develop this additional API for testing; test group believes that this should also be sufficient for their needs.